



**TUGAS AKHIR - 145565**

**PERANCANGAN APLIKASI *MONITORING* SISTEM  
*TRACKING PHOTOVOLTAIC* SEBAGAI SUMBER ENERGI  
LISTRIK PADA *SMART CLASSROOM* DENGAN  
KOMUNIKASI DATA BERBASIS *WIRELESS***

**ERGI QIBAR NUGRAHA**  
10 51 15 000 00 078

Dosen Pembimbing 1:  
Dr.Ir. Purwadi Agus Darwito, M.Sc  
NIP. 19620822 198803 1 001

Dosen Pembimbing 2:  
Murry Raditya, S.T,M.T  
NPP. 1988 2017 11055

DEPARTEMEN TEKNIK INSTRUMENTASI  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember  
Surabaya  
2018





**TUGAS AKHIR - 145565**

**PERANCANGAN APLIKASI MONITORING SISTEM  
TRACKING PHOTOVOLTAIC SEBAGAI SUMBER ENERGI  
LISTRIK PADA SMART CLASSROOM DENGAN  
KOMUNIKASI DATA BERBASIS WIRELESS**

**ERGI QIBAR NUGRAHA**  
**10 51 15 000 00 078**

**Dosen Pembimbing 1:**  
**Dr.Ir. Purwadi Agus Darwito, M.Sc**  
**NIP. 19620822 198803 1 001**

**Dosen Pembimbing 2:**  
**Murry Raditya, S.T,M.T**  
**NPP. 1988 2017 11055**

**DEPARTEMEN TEKNIK INSTRUMENTASI**  
**Fakultas Vokasi**  
**Institut Teknologi Sepuluh Nopember**  
**Surabaya**  
**2018**

## LEMBAR PENGESAHAN I

### **“PERANCANGAN APLIKASI MONITORING SISTEM TRACKING PHOTOVOLTAIC SEBAGAI SUMBER ENERGI LISTRIK PADA SMART CLASSROOM DENGAN KOMUNIKASI DATA BERBASIS WIRELESS”**

#### **TUGAS AKHIR**

Oleh :

**ERGI QIBAR NUGRAHA**  
**NRP. 105 1 15 000 00 078**

**Surabaya, 27 Juli 2018**  
**Mengetahui / Menyetujui**

Dosen Pembimbing I

Dr. Ir. Purwadi Agus Darwito, M.Sc  
NIP. 19620822 198803 1 001

Dosen Pembimbing II

Murry Raditya, S.T., M.T  
NPP. 1998 2017 11055



## **LEMBAR PENGESAHAN II**

### **“PERANCANGAN APLIKASI MONITORING SISTEM TRACKING *PHOTOVOLTAIC* SEBAGAI SUMBER ENERGI LISTRIK PADA *SMART CLASSROOM* DENGAN KOMUNIKASI DATA BERBASIS *WIRELESS*”**




#### **TUGAS AKHIR**

Diajukan Untuk Memperoleh salah satu syarat  
Untuk memperoleh gelar Ahli Madya  
Pada  
Program studi D3 Teknologi Instrumentasi  
Departemen Teknik Instrumentasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember

Oleh :

**ERGI QIBAR NUGRAHA**  
**NRP.105 1 15 000 00 078**

Disetujui oleh Tim Penguji Tugas Akhir :

1. Dr. Ir. Purwadi Agus Darwito, M.Sc.  Pembimbing I
2. Murry Raditya, S.T, M.T.  Pembimbing II
3. Ir. Tutug Dhanardono, M.T.  Penguji

**SURABAYA**  
**2018**

**PERANCANGAN APLIKASI MONITORING SISTEM  
TRACKING *PHOTOVOLTAIC* SEBAGAI SUMBER  
ENERGI LISTRIK PADA *SMART CLASSROOM* DENGAN  
KOMUNIKASI DATA BERBASIS *WIRELESS***

**Nama** : Ergi Qibar Nugraha  
**NRP** : 10511500000078  
**Departemen** : Teknik Instrumentasi FV-ITS

**ABSTRAK**

Energi adalah kemampuan untuk melakukan pekerjaan. Tidak diragukan lagi bahwa energi surya adalah salah satu sumber energi yang ramah lingkungan dan sangat menjanjikan pada masa yang akan datang, karena tidak ada polusi yang dihasilkan selama proses konversi energi, dan juga sumber energinya banyak tersedia di alam. Kita dapat mengkonversi energy matahari menjadi energy listrik menggunakan fotovoltaiik. Namun karena sumber energi matahari yang tidak bisa menyinari utuh seharian penuh maka muncul inovasi bernama solar tracking, yang diharapkan mampu mengubah energi matahari menjadi energi listrik secara maksimal, untuk mengetahui bagaimana hasil energy yang didapatkan maka dibuat sistem monitoring.

**Kata kunci:**energi surya, fotovoltaiik, Solar tracking, monitoring

***DESIGN OF A MONITORING APPLICATION SYSTEM  
TRACKING PHOTOVOLTAIC AS A SOURCE OF ENERGY  
ON SMART CLASSROOM BASED ON WIRELESS  
COMMUNICATION***

**Name : Ergi Qibar Nugraha**  
**NRP : 10511500000078**  
**Department : Instrument Engineering FV-ITS**

***ABSTRACT***

Energy is the ability to do work. There is no doubt that solar energy is one of the most environmentally friendly and promising sources of energy in the future, since no pollution is generated during the energy conversion process, and also its energy source is widely available in nature. We can convert solar energy into electrical energy using photovoltaic. However, because the source of solar energy that can not illuminate intact all day then came the innovation called solar tracking, which is expected to convert solar energy into electrical energy to the maximum, to find out how the energy obtained then made the monitoring system.

**Keywords :** solar energy, photovoltaic, Solar tracking, monitoring

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas berkat, rahmat dan kebesaran-Nya sehingga penulis dapat menyelesaikan laporan tugas akhir yang berjudul “PERANCANGAN APLIKASI MONITORING SISTEM TRACKING *PHOTOVOLTAIC* SEBAGAI SUMBER ENERGI LISTRIK PADA *SMART CLASSROOM* DENGAN KOMUNIKASI DATA BERBASIS *WIRELESS*”

Selama menyelesaikan Tugas Akhir ini penulis telah banyak mendapatkan bantuan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih ang sebesar-besarnya kepada :

1. Keluarga tercinta, orangtua penulis terima kasih atas dorongan semangatnya, bantuan dan dukungannya selama ini sehingga laporan ini dapat selesai dengan baik.
2. Amalia Puruhita, teman penulis terimakasih atas segala semangat dan dukungannya.
3. Bapak Dr. Ir. Purwadi Agus Darwito, MSc selaku kepala departemen teknik instrumentasi ITS, Surabaya.
4. Bapak Dr. Ir. Purwadi Agus Darwito, MSc selaku dosen pembimbing pertama Tugas Akhir.
5. Bapak Murry Raditya, S.T,M.T selaku dosen pembimbing kedua Tugas Akhir.
6. Rekan-rekan team tugas akhir *Smart Classroom* atas kekompakan dan kerjasamanya.
7. Teman-teman D3 teknik Instrumentasi angkatan 2015 yang membantu dan mensupport selama kegiatan Tugas Akhir berlangsung.
8. Chintya Sagita Goestien dan Heru atas bantuan pengerjaan tugas akhir ini.
9. Seluruh karyawan dan staff Departemen Teknik Instrumentasi yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa kesempurnaan hanya milik Allah SWT. Oleh karena itu, penulis sangat berterimakasih atas segala masukan, kritik dan saran yang membangun dari pembaca



agar laporan ini menjadi lebih baik lagi untuk di kemudisn hari. Demikian laporan ini penulis buat, semoga laporan ini dapat memberikan manfaat selain bagi penulis sendiri, dan bagi pembaca sekalian.

Surabaya, 27 Juli 2018  
Penulis

Ergi Qibar Nugraha  
NRP. 10 5115 00000 078

## DAFTAR ISI

LEMBAR PENGESAHAN I .....	ii
LEMBAR PENGESAHAN II .....	iii
ABSTRAK .....	iv
<i>ABSTRACT</i> .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	ix
BAB I .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan .....	2
1.4 Batasan Masalah .....	3
1.5 Sistematika Laporan .....	3
BAB II .....	5
2.1 Fotovoltaik .....	5
2.2 Solar Tracking .....	6
2.5 Sensor .....	9
2.6 Mikrokontroler ARM STM32F401 (Nucleo) .....	11
2.7 Visual C# .....	11
BAB III .....	15
3.1 Flowchart Penelitian .....	15
3.2 Perancangan Sistem .....	18
3.2.1 Perancangan Hardware .....	18
3.2.2 Perancangan Software .....	21
BAB IV .....	27
4.1 Hasil Pengerjaan Sistem .....	27
4.1.1 Pengujian Alat Dengan Data Pada SD Card .....	38
4.1.2 Pengujian Alat Dengan Variasi Jarak .....	41
4.2 Pembahasan .....	45
BAB V .....	47
5.1 Kesimpulan .....	47
5.2 Saran .....	47
DAFTAR PUSTAKA .....	49

## DAFTAR GAMBAR

Gambar 2. 1 Kurva I-V pada modul PV Kyocera KC200GT: (a) Pada beberapa level radiasi, (b) Pada beberapa suhu sel .....	5
Gambar 2. 2 Rangkaian sel dioda model satu diode .....	6
Gambar 2. 3 Panel Surya bertingkat .....	7
Gambar 2. 4 Ilustrasi Pergerakan Panel Surya.....	7
Gambar 2. 5 Rangkaian sensor ACS712 .....	10
Gambar 2. 6 Diagram Blok Sensor.....	11
Gambar 2. 7 STM32F401RE.....	11
Gambar 2. 8 Logo Visual C#.....	12
Gambar 2. 9 Arsitektur SD Card .....	13
Gambar 2. 10 Modul 3dr Telemetry .....	14
 Gambar 3. 1 Blok Diagram Keseluruhan Sistem.....	15
Gambar 3. 2 Flowchart Penelitian .....	18
Gambar 3. 3 3DR Config .....	20
Gambar 3. 4 Pseudocode SD Card .....	21
Gambar 3. 5 Header C#.....	22
Gambar 3. 6 Pseudocode variable yang akan ditampilkan pada Aplikasi .....	22
Gambar 3. 7 Pseudocode Start button.....	23
Gambar 3. 8 Pseudocode Serial Port .....	24
Gambar 3. 9 Pseudocode Grafik Update .....	25
Gambar 3. 10 Pseudocode Stop Button .....	25
 Gambar 4. 1 Diagram Blok Sistem Monitoring Tracking Fotovoltaik .....	27
Gambar 4. 2 Hasil Perancangan Hardware .....	28
Gambar 4. 3 Hasil Perancangan Software .....	28
Gambar 4. 4 Pembacaan Data Pada HMI .....	39
Gambar 4. 5 Pembacaan Data SD Card.....	40
Gambar 4. 6 Pembacaan Sumber Tegangan oleh Multimeter .....	40
Gambar 4. 7 Pembacaan Data Dengan Jarak 1 Meter.....	41
Gambar 4. 8 Pembacaan Dengan Jarak 5 Meter .....	42
Gambar 4. 9 Pembacaan Dengan Jarak 15 Meter .....	42

Gambar 4. 10 Pembacaan Dengan Jarak 25 Meter .....	43
Gambar 4. 11 Pembacaan Dengan Jarak 50 Meter .....	44
Gambar 4. 12 Pembacaan Dengan Jarak 100 Meter .....	44

(Halaman ini sengaja dikosongkan)



# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Energi adalah kemampuan untuk melakukan pekerjaan. Energi adalah daya yang dapat digunakan untuk melakukan berbagai proses kegiatan meliputi energi mekanik, panas, dan lain – lain. Oleh karena itu, hampir semua perselisihan di dunia ini, berpangkal pada perebutan sumber energi. Ada beberapa energi alam sebagai energi alternatif yang bersih, tidak berpolusi, aman dan persediaannya tidak terbatas yang dikenal dengan energi terbarukan. [1]

Tidak diragukan lagi bahwa energi surya adalah salah satu sumber energi yang ramah lingkungan dan sangat menjanjikan pada masa yang akan datang, karena tidak ada polusi yang dihasilkan selama proses konversi energi, dan juga sumber energinya banyak tersedia di alam [2]

*Solar cell* konvensional bekerja menggunakan prinsip p-n *junction*, yaitu *junction* antara semikonduktor tipe-p dan tipe-n. Semikonduktor tipe-n mempunyai Kelebihan elektron (muatan negatif) sedangkan tipe-p mempunyai kelebihan *hole* (muatan positif) dalam struktur atomnya. Prinsip kerja dari *solar cell* adalah dengan memanfaatkan efek *photovoltaic* (PV, *photo* = cahaya dan *voltaic* = listrik) dalam bentuk sel surya atau *solar cell* yang terbuat dari silikon berkrystal tunggal. Cahaya matahari yang membawa energi akan diterima sel dan diserap ke dalam semi konduktor sehingga mempengaruhi elektron yang ada di dalamnya. Elektron bebas akan digerakkan oleh energi tersebut sehingga mengalir ke arah tertentu. Aliran elektron ini yang dinamakan sebagai arus listrik [3]

Sistem kendali *tracking* sinar matahari merupakan salah satu pengembangan teknologi dalam rangka konservasi energi dengan memanfaatkan energi matahari. Sistem kendali *tracking* dibuat untuk diterapkan lebih lanjut pada sebuah perangkat panel surya sebagai alat pengumpul energi matahari sehingga posisi panel surya dinamis [4]

Agar energi dari hasil konversi sinar matahari dapat di pantau besaran fisisnya berupa daya maka sistem monitoring sangat

diperlukan. Berbagai penelitian telah dilakukan untuk monitoring sistem tenaga listrik [5], salah satunya telah dilakukan penelitian pada sistem tenaga dengan menggunakan mikrokontroler BCM2835 yang digunakan sebagai referensi oleh penulis [6].

Departemen Teknik Instrumentasi adalah salah satu departemen yang sedang mengembangkan smartclassroom. Sebelumnya telah ada penelitian tentang smart classroom berbasis kartu yang telah dilakukan oleh Gintoro, Adrian Tjahjana, Dedy Irawan, William. Namun bila di departemen teknik instrumentasi smart classroom yang sedang dikembangkan yaitu dengan sumber tenaga mayoritas dari energy alternative atau terbarukan salah satunya adalah tenaga matahari, diperlukan panel surya yang mumpuni guna mendapatkan hasil yang maksimal.

Selain hardware software juga diperlukan untuk mengintegrasikan sebuah sistem agar lebih mudah memantau atau monitoring Arus daya dan tegangan. Dengan menggunakan komunikasi data nirkabel atau biasa disebut *wireless* dan menggunakan software Visual Studio maka diusunglah tugas akhir dengan judul “PERANCANGAN APLIKASI MONITORING SISTEM TRACKING *PHOTOVOLTAIC* SEBAGAI SUMBER ENERGI LISTRIK PADA *SMART CLASSROOM* DENGAN KOMUNIKASI DATA BERBASIS *WIRELESS*”. Yang diharapkan mampu mendapatkan hasil yang akurat tentang hasil keluaran dari system solar *tracking* sehingga dapat menjadi sumber utama energy terbarukan pada smart classroom ruang kelas A101 di Departemen Teknik Instrumentasi ITS dan sebagai acuan penelitian-penelitian berikutnya.

## **1.2 Rumusan Masalah**

Adapun permasalahan dalam tugas akhir ini adalah:

1. Bagaimana mendesain sistem monitoring hasil pengukuran arus tegangan dan daya sistem tracking Photovoltaic secara realtime?
2. Bagaimana merancang sistem monitoring sistem tracking photovoltaic berbasis wireless ?

## **1.3 Tujuan**

Tujuan utama dari rancang bangun alat ini adalah untuk memenuhi mata kuliah tugas akhir sebagai syarat kelulusan dari



program studi diploma 3 metrologi dan instrumentasi, serta untuk memberikan solusi pada rumusan masalah yaitu :

1. Mendesain sistem monitoring hasil pengukuran arus tegangan dan daya sistem tracking Photovoltaic secara realtime.
2. Menghasilkan sistem monitoring tracking sistem photovoltaic berbasis wireless.

#### **1.4 Batasan Masalah**

Adapun batasan ruang lingkup dari penelitian tugas akhir ini yaitu hanya membahas mengenai sistem *Tracking Photovoltaic* untuk smart classroom A101 Teknik Instrumentasi ITS, module wireless menggunakan RF(*radio frequency*).

#### **1.5 Sistematika Laporan**

Adapun sistematika dari laporan ini yaitu terdiri 5 bab, dimana masing-masing bab tersusun dari beberapa sub-bab, yaitu:

1. Bab 1 yaitu pendahuluan menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, dan sistematika laporan
2. Bab 2 yaitu tinjauan pustaka menjelaskan solar tracking, sensor, motor stapper, momen inersia, dan torsi.
3. Bab 3 terdiri dari metodologi
4. Bab 4 terdiri dari analisa data dan pembahasan
5. Bab 5 yaitu penutup yang terdiri atas kesimpulan dan saran dari Tugas Akhir

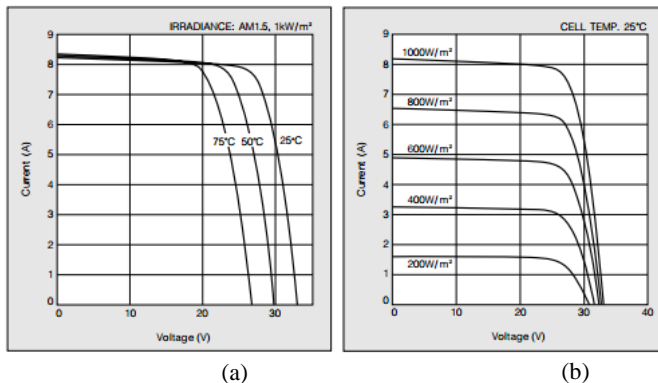
(Halaman ini sengaja dikosongkan)

## BAB II

### TINJAUAN PUSTAKA

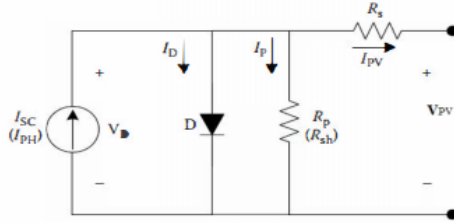
#### 2.1 Fotovoltaik

Berdasarkan judul diatas, Fotovoltaik adalah sumber tenaga yang akan di *implementasikan* pada *smart classroom*. Pengertian Fotovoltaik adalah suatu sistem yang bekerja dengan prinsip efek fotovoltaik yang mentransfer radiasi matahari menjadi energi listrik. Daya keluaran modul fotovoltaik bergantung pada dua faktor yaitu suhu sel dan radiasi matahari. Dua faktor tersebut tidak dapat dipisahkan antar satu dengan yang lain, karena faktor suhu sel dapat mempengaruhi tegangan keluaran modul PV, sedangkan radiasi matahari mempengaruhi arus keluaran modul PV. Setiap modul PV memiliki spesifikasi yang berbeda tergantung tipenya, sehingga karakteristik kurva I-V modul PV pun juga berbeda-beda. Berikut ini ialah kurva I-V pada modul PV Kyocera KC200GT [7] :



**Gambar 2. 1** Kurva I-V pada modul PV Kyocera KC200GT: (a) Pada beberapa level radiasi, (b) Pada beberapa suhu sel

Untuk pemodelan sel surya, model satu dioda telah diusulkan, dan beberapa peneliti telah mempelajari cara mengekstrak parameter untuk model. Berikut adalah gambar rangkaian sel dioda model satu diode:



**Gambar 2. 2** Rangkaian sel dioda model satu diode

Berdasarkan rangkaian diatas dapat diketahui beberapa rumus berikut:

$$I_D = I_0 \left[ e^{\frac{V_{PV}}{\alpha V_T}} - 1 \right] \quad (1)$$

$$I_{PV} = I_{SC} - I_0 \left( e^{\frac{q(V_{PV} + I_{PV} \cdot R_s)}{\alpha K T}} - 1 \right) - \frac{V_{PV} + I_{PV} \cdot R_s}{R_{sh}} \quad (2)$$

$$I_{PV} = I_{SC} - I_D \quad (3)$$

$$V_{PV} = \alpha V_T \ln \left[ \frac{(I_{SC} - I_{PV})}{I_0} + 1 \right] \quad (4)$$

Dimana:

$I_D$  = Arus Dioda

$I_0$  = Arus Saturasi Balik

$V_{PV}$  = Tegangan Panel

$V_T$  = Tegangan Suhu

$I_{PV}$  = Arus yang melalui Panel

$I_{SC}$  = Arus Foton

$\alpha$  = Faktor ideal PV

$V_T = kT/q$

$K$  = Konstanta Boltzmann (J/K) ( $1.380 \times 10^{-23}$ )

$T$  = Suhu sel PV (K)

$Q$  = daya elektron (C) ( $-1.602 \times 10^{-19}$ )

(5)

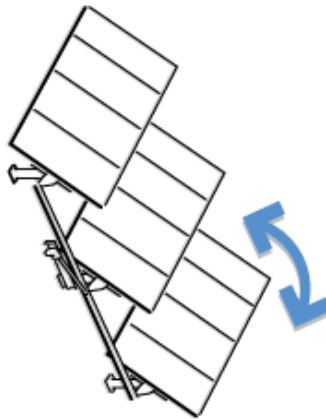
## 2.2 Solar Tracking

Pada penelitian ini menggunakan sumbu vertical. Sumbu rotasi untuk pelacak sumbu vertikal terhadap tanah. Pelacak ini

berputar dari Timur ke Barat sepanjang hari. Pelacak semacam itu lebih efektif pada lintang tinggi daripada pelacak sumbu horisontal. Tata letak lapangan harus mempertimbangkan naungan untuk menghindari kerugian energi yang tidak perlu dan untuk mengoptimalkan pemanfaatan lahan. Juga pengoptimalan untuk pengepakan padat terbatas karena sifat shading selama setahun. Pelacak sumbu vertikal biasanya memiliki permukaan modul yang berorientasi pada sudut berkenaan dengan poros rotasi. Sebagai trek modul, ia menyapu kerucut yang berputar simetris di sekitar sumbu rotasi [8].



**Gambar 2. 3** Panel Surya bertingkat



**Gambar 2. 4** Ilustrasi Pergerakan Panel Surya

### 2.3 Sistem Monitoring

Monitoring adalah pemantauan yang dapat dijelaskan sebagai kesadaran (awareness) tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu. *Monitoring* akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu, pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa terhadap proses berikut objek atau untuk mengevaluasi kondisi atau kemajuan menuju tujuan hasil manajemen atas efek tindakan dari beberapa jenis antara lain tindakan untuk mempertahankan manajemen yang sedang berjalan. Berdasarkan judul diatas system monitoring bertujuan untuk memantau bagaimana perubahan nilai besaran fisis yang ada pada plant. Sehingga data dapat dijadikan acuan yang tepat [9].

Sistem monitoring ter dir dari beberapa proses yaitu sebagai berikut:

1. Sensing = yaitu untuk mengukur besaran fisis yang diinginkan seperti tegangan atau arus, pada umumnya untuk mengukur besaran fisis digunakan sensor/transduser yang akan mengubah besaran fisis menjadi data analog.
2. Prosesing = yaitu proses mengubah data analog menjadi digital atau pada umumnya disebut ADC(analog to digital converter) mikrokontroller berperan penting dalam menjalankan ADC ini.
3. Recording : recording adalah penyimpanan data yang telah dibaca agar kita dapat menjadikan acuan apabila ingin mengolah data tersebut
4. Display : Display merupakan proses terakhir yaitu menampilkan data yang telah diolah oleh controller dengan tujuan agar manusia dapat membaca besaran fisis dengan mudah.

### 2.4 Human Machine Interfacing(HMI)

HMI (Human Machine Interface) adalah sistem yang menghubungkan antara manusia dan teknologi mesin. HMI dapat berupa pengendali dan visualisasi status baik dengan manual

maupun melalui visualisasi komputer yang bersifat real time. Sistem HMI biasanya bekerja secara online dan real time dengan membaca data yang dikirimkan melalui I/O port yang digunakan oleh sistem controller-nya. Port yang biasanya digunakan untuk controller dan akan dibaca oleh HMI antara lain adalah port com, port USB, port RS232 dan ada pula yang menggunakan port serial. Tugas dari HMI (Human Machine Interface) yaitu membuat visualisasi dari teknologi atau sistem secara nyata. Sehingga dengan desain HMI dapat disesuaikan sehingga memudahkan pekerjaan fisik. Tujuan dari HMI adalah untuk meningkatkan interaksi antara mesin dan operator melalui tampilan layar komputer dan memenuhi kebutuhan pengguna terhadap informasi sistem. HMI dalam industri manufacture berupa suatu tampilan GUI (Graphic User Interface) pada suatu tampilan layar komputer yang akan dihadapi oleh operator mesin maupun pengguna yang membutuhkan data kerja mesin. HMI terdapat berbagai macam visualisasi untuk Monitoring dan data mesin yang terhubung secara online dan real time. HMI akan memberikan suatu gambaran kondisi mesin yang berupa peta mesin produksi dapat dilihat bagian mesin mana yang sedang bekerja. Pada HMI juga terdapat visualisasi pengendali mesin berupa tombol, slider, dan sebagainya yang dapat difungsikan untuk mengendalikan mesin sebagaimana mestinya. Selain itu dalam HMI juga ditampilkan alarm jika terjadi kondisi bahaya dalam sistem. Sebagai tambahan, HMI juga menampilkan data-data rangkuman kerja mesin termasuk secara grafik [10].

## **2.5 Sensor**

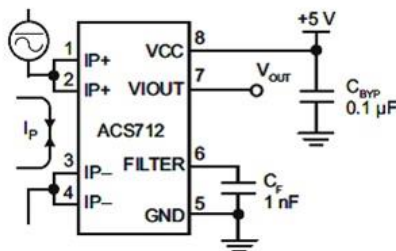
Sensor arus adalah suatu alat yang mengukur jumlah arus pada alat elektronik. Sensor arus biasanya terdiri dari rangkaian elektronik yang mengubah jumlah arus menjadi satuan listrik. Sensor arus yang biasa digunakan adalah Current Transducer . Sensor arus bekerja dengan membaca arus yang melalui tembaga yang ada di dalamnya menghasilkan medan magnet yang ditangkap oleh integrated Hall IC dan diubah menjadi tegangan proporsional. Berdasarkan judul diatas digunakan sensor tegangan dan sensor arus ACS712.

### 2.5.1 Sensor Arus

ACS712 merupakan suatu IC terpaket yang mana berguna sebagai sensor arus menggantikan transformator arus yang relatif besar dalam hal ukuran. Pada prinsipnya ACS712 sama dengan sensor efek hall lainnya yaitu dengan memanfaatkan medan magnetik disekitar arus kemudian dikonversi menjadi tegangan yang linier dengan perubahan arus [11]. Nilai variabel dari sensor ini merupakan input untuk mikrokontroler yang kemudian diolah. Keluaran dari sensor ini masih berupa sinyal tegangan AC, agar dapat diolah oleh mikrokontroler maka sinyal tegangan AC ini di searahkan oleh rangkaian penyearah.

Memiliki Rumus :

$$\text{Arus} = ((\text{ADC\_Arus}/V_{\text{reff}}) * \text{Bit}) - 2.5) / 100$$



**Gambar 2. 5** Rangkaian sensor ACS712

### 2.5.2 Sensor Tegangan

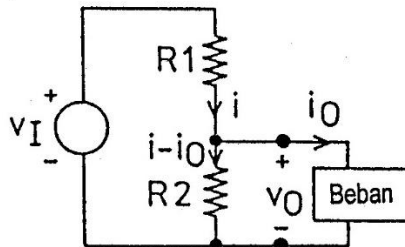
Sensor tegangan yang digunakan merupakan sebuah modul sensor tegangan yang menggunakan prinsip pembagi tegangan [12]. Modul ini dapat mengurangi tegangan input hingga 5 kali dari tegangan asli. Tegangan analog input maksimum mikrokontroler yaitu 5 volt, sehingga modul tegangan dapat diberi masukan tidak melebihi 5 X 5 Volt atau sebesar 25 Volt. Untuk dapat mengukur tegangan dengan benar maka digunakan rumus:

$$\text{Tegangan} = (\text{ADC}/V_{\text{reff}}) * \text{bitkontroler}$$

Dimana ADC adalah data pembacaan sinyal analog dari plant dan akan dibagi oleh Vreff dari kontroller, bergantung dengan jenis



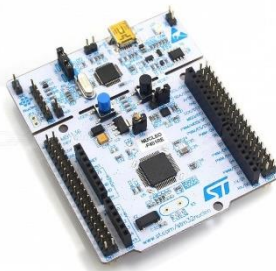
kontrollernya maka nilai  $V_{\text{reff}}$  juga akan berbeda-beda. Pada tugas akhir ini  $V_{\text{reff}}$  yang digunakan sebesar 5volt dan 8 bit yaitu 1024.



**Gambar 2. 6** Diagram Blok Sensor

## 2.6 Mikrokontroler ARM STM32F401 (Nucleo)

Development board berbasis mikrokontroler ARM Cortex-M4 STM32F401RE. Memiliki dukungan terhadap arduino-shield, serta dapat diprogram secara web-based melalui mbed•org, atau melalui software yang mendukung chip mikrokontroler STM32F401, seperti Keil MDK ARM dan yang lainnya. STM32F401 bekerja pada frekuensi clock max 84 MHz, 0-wait state, memiliki ruang flash 512 KB dan SRAM 96 KB [13].



**Gambar 2. 7** STM32F401RE

## 2.7 Visual C#

C#(dibaca:Csharp)merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. Bahasa

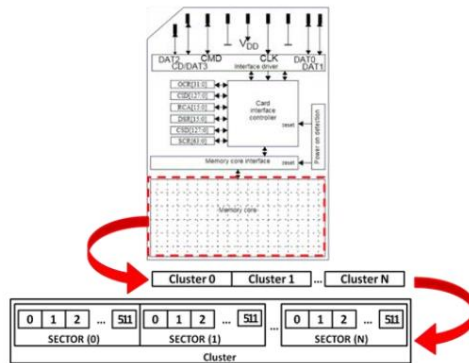
pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain) dengan beberapa penyederhanaan. Menurut standar ECMA-334 C# Language Specification, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan angka # (U+0023). Tanda pagar # yang digunakan memang bukan tanda kres dalam seni musik (U+266F), dan tanda pagar # (U+0023) tersebut digunakan karena karakter kres dalam seni musik tidak terdapat di dalam keyboard standar.



**Gambar 2. 8** Logo Visual C#

## **2.8 SD Card**

Gambar 2.10. Arsitektur SD card Seluruh data file terletak pada Memory core. Memory core terdiri dari beberapa cluster dan setiap cluster berisi beberapa sektor. Masing-masing sektortersusun dari kumpulan byte dan setiap byte memiliki 8 bit lokasi memori. Format data SD card umumnya menggunakan format File Allocation Table (FAT). FAT12 digunakan untuk kapasitas lebih kecil dari 16 MB. FAT16 digunakan untuk kapasitas 32 MB hingga 2 GB. FAT32 digunakan untuk kapasitas di atas 2 GB (SDHC).



**Gambar 2. 9** Arsitektur SD Card

.Hal pertama yang harus dilakukan adalah membaca parameter SD card dengan urutan langkah sebagai berikut:

1. Mengirimkan perintah Reset dan Init ke SD card.
2. Membaca Master Boot Record (berada di sektor 0) untuk mengetahui lokasi Boot Sector. Lalu Boot Sector dibaca secara keseluruhan.
3. Nilai pada alamat-alamat tertentu dari Boot Sector diambil dan dihitung sehingga didapat parameter antara lain: alamat FAT Region, alamat Root Directory, alamat Data Region, jumlah sector per cluster, tipe FAT, dan kapasitas SD card.

Setelah membaca parameter SD card, maka langkah selanjutnya adalah menulis data pada SD card. Urutan langkahnya adalah sebagai berikut:

1. Mengirimkan perintah Reset dan Init ke SD card.
2. Selanjutnya program akan menulis Nama Kartu dan Nama File pada Root Directory. Nilai Root Directory akan dibaca untuk mengetahui posisi sector awal untuk file. Program akan menulis tabel FAT pada FAT Region [14].

## 2.9 3DR Telemetry Modul

3DR telemetry modul adalah sebuah modul untuk berkomunikasi berbasis Radio Frekuensi. Memiliki ukuran yang kecil dan jarak yang jauh yaitu 500m. memiliki kecepatan transfer hingga 250 KBPS di udara dan memiliki ukuran yang kecil

sehingga mudah dibawa. Memiliki 4 pin yaitu VCC, GND, TX dan RX, serta memiliki 2 lampu penanda yaitu merah dan hijau. Tiap paket 3DR telemetry terdiri dari satu modul APM sebagai pengirim dan modul Ground sebagai penerima [15].



**Gambar 2. 10** Modul 3dr Telemetry

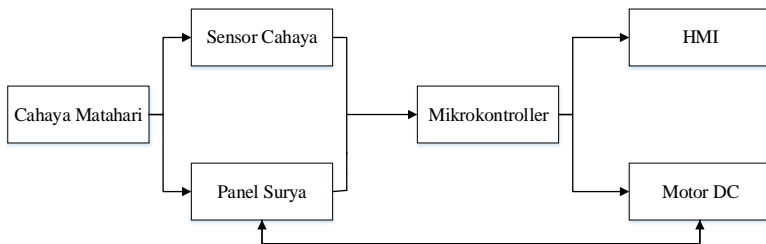
## **2.10 Mbed OS**

Mbed OS merupakan sebuah operating system yang dirancang secara khusus untuk mikrokontroler berbasis ARM Cortex-M 32-bit. Dalam aplikasinya Mbed OS sudah banyak mendukung banyak mikrokontroller dari berbagai vendor. Mbed OS sendiri dibuat dengan menggunakan bahasa C dan C++ sehingga program yang dikembangkan dengan OS ini lebih mudah dan dapat diexport pada beberapa IDE khusus pengembangan mikrokontroler. Mbed OS juga dapat meng-import program dari IDE lain ke dalam sistemnya. Untuk melakukan tugas tersebut Mbed OS ini harus berjalan secara Online [16]

## BAB III METODOLOGI PENELITIAN

### 3.1 Flowchart Penelitian

Mengacu pada judul tugas akhir ini perancangan tugas akhir ini memiliki keterkaitan dengan sistem yang lainnya. Adapun blok diagram sistem keseluruhan alat adalah sebagai berikut :



**Gambar 3. 1** Blok Diagram Keseluruhan Sistem

Dari blok diagram keseluruhan alat diatas terdapat HMI bagian yang akan dibahas, yang dalam ruang lingkup lebih luas disebut sistem monitoring monitoring system tracking fotovoltaiik sebagai sumber energy *Smart Classroom* yang juga menggunakan komunikasi data berbasis wireless.. Adapun pengerjaan alat yang nantinya disederhanakan dalam bentuk diagram alir seperti pada Gambar 3.2 yang dibagi menjadi dua kategori yaitu perancangan hardware dan software dijelaskan dalam tahapan-tahapan sebagai berikut :

A. Studi Literatur dibagi menjadi dua yaitu pengukuran arus daya dan tegangan serta penyimpanan data logger berbasis STM32F401RE dan frekuensi komunikasi data berbasis wireless antara HMI dan STM32F401RE.

B. kemudian dilakukan perumusan masalah tentang kedua bagian diatas yaitu pengukuran arus daya dan tegangan serta penyimpanan data logger berbasis STM32F401RE dan sistem komunikasi data antara HMI dan mikrokontroler berbasis wireless.

C. Perancangan program *monitoring* arus daya dan tegangan serta

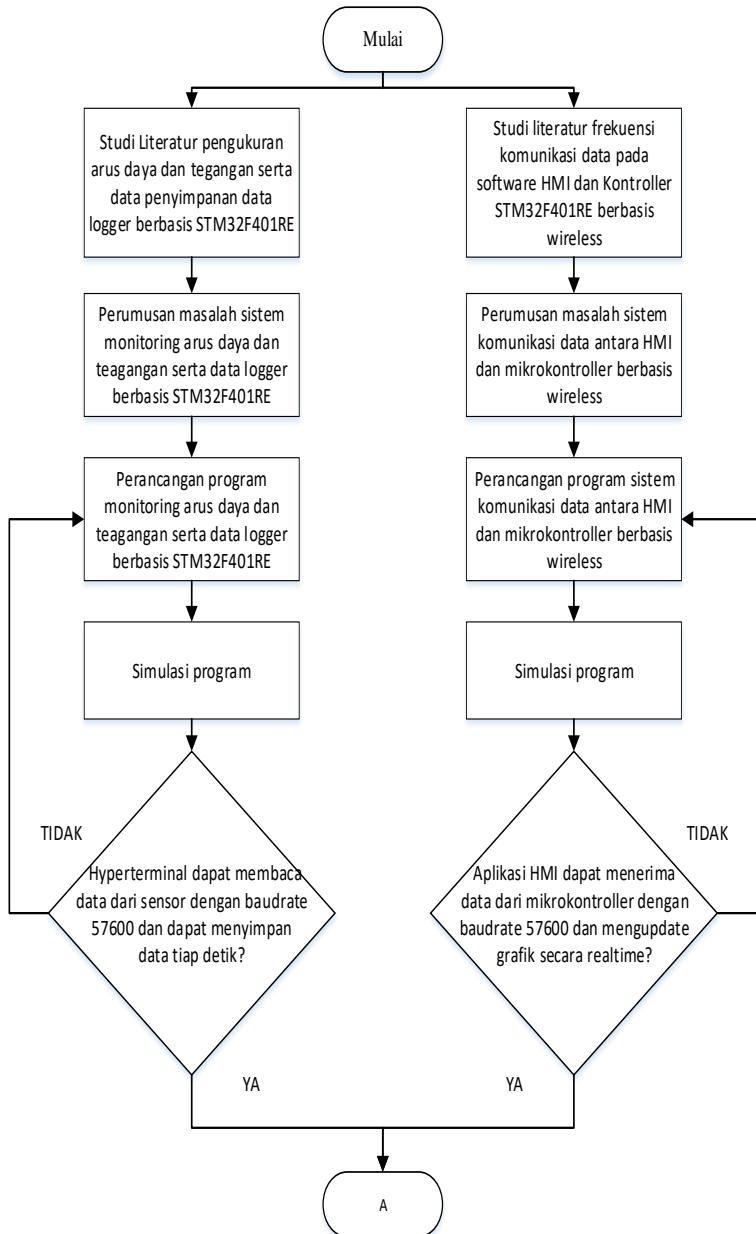
penyimpanan data logger berbasis STM32F401RE menggunakan software keil uvision, dan Perancangan program sistem komunikasi data antara HMI dan mikrokontroller, HMI dirancang menggunakan software Visual Studio menggunakan Bahasa pemrograman C# dan mikrokontroller deprogram dengan Bahasa C pada software Keil uVision.

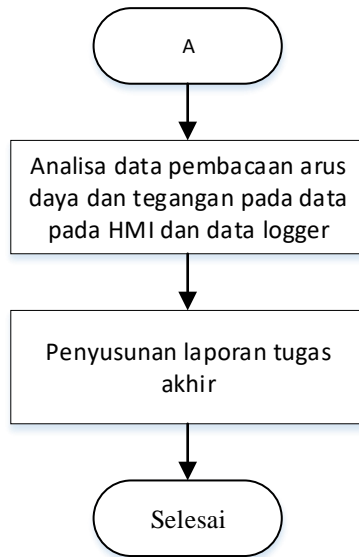
D. Simulasi kedua program yang telah dirancang

E. Apabila hyperterminal dapat membaca hasil pengukuran sensor berupa angka pada baudrate 57600 maka perancangan hardware dianggap berhasil dan HMI dapat menerima data dari Mikrokontroller dengan baudrate 57600 dan mneng update grafik secara realtime maka akan dilanjutkan ke tahap berikutnya.

F. Hasil pengambilan data di analisa

G. Penyusunan laporan





**Gambar 3. 2** Flowchart Penelitian

### 3.2 Perancangan Sistem

Pada perancangan sistem ini dijelaskan dalam dua poin yaitu hardware dan software sebagai berikut

#### 3.2.1 Perancangan Hardware

Dalam perancangan Hardware terdapat 3 hal penting yang harus dikerjakan sebagai berikut:

##### A.Sensor Arus

Sensor arus yang digunakan adalah Sensor arus ACS 712 dengan range 0-20 Ampere sensor ini dipasang Secara seri dari sumber arus. Kemudian pin Analogin Disambungkan ke controller yaitu pin PC\_5. agar dapat menghitung arus data dari sensor harus diolah terlebih dahulu pada mikrokontroler dengan program sebagai berikut

```

Arus_ADC = analog_Arus.read();
Arus = del_v/100 ;
del_v = ((Arus_ADC/5)*1024)-2.5;
pc.printf("A%.2f\n", Arus);
  
```



Dari program diatas diketahui bahwa untuk mendapatkan nilai arus dibuat variable baru bernama `del_V` untuk atau delta v kemudian dirumuskan bahwa nilai `del_V` sama dengan `Arus_ADC` dikali 3.3 yang merupakan `vref` dari STM401RE kemudian dibagi jumlah bit dari mikro yaitu 12 bit menjadi 4096 dan dikurangi 2.5 yang merupakan nilai tegangan minimum. Setelah `del_V` didapat dicari nilai arus yaitu nilai `del_V` dikali sensitivitas dari sensor ACS712 20 Ampere yaitu 0.1 Volt.

### B. Sensor Tegangan

Sensor tegangan dipasang secara parallel terhadap sumber tegangan. Module sensor tegangan menggunakan prinsip kerja pembagi tegangan. Memiliki range pembacaan 0-25 VDC dan output 0-5 VDC, Kemudian pin Analogin pada modul disambungkan ke kontroler pada pin PC\_4. Untuk mendapatkan hasil pembacaan tegangan mikrokontroler hasil deprogram terlebih dahulu agar data Analog dari sensor berubah menjadi digital

```
Tegangan_ADC = analog_Tegangan.read();
Tegangan=((Tegangan_ADC/5)*1024/10)+E1;
pc.printf("V%.2f \n", Tegangan);
```

Dari program diatas dapat diketahui bahwa untuk mendapatkan hasil pembacaan Tegangan digunakan rumus  $Tegangan = (Tegangan\_ADC/3.3)*4095$ . Dimana tegangan ADC adalah pembacaan sinyal analog dari sensor tegangan dibagi 3 yang merupakan `Vcc` lalu dikali 4096 yang merupakan jumlah dari bit mikrokontroller yang digunakan(STM32F401RE 12bit)

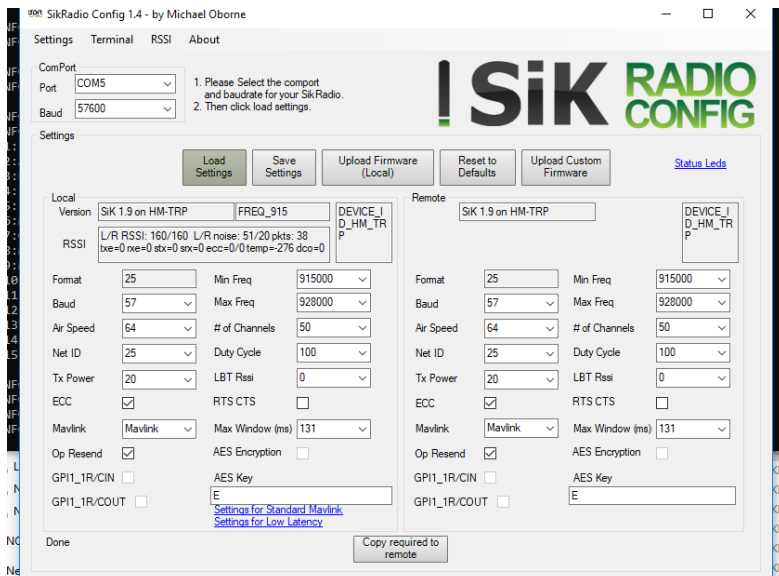
### C. 3DR Telemetry Wireless Module 915 MHZ

Untuk komunikasi data antara Kontroller dan HMI menggunakan module 3dr telemetry module yang merupakan modul komunikasi data berbasis wireless menggunakan Radio Frequency. modul ini memiliki konfigurasi untuk bias berkomunikasi yaitu baudrate diatur sebesar 57600 dan pinout TX, RX disambungkan ke kontroler untuk modul transceiver yaitu pada pin dengan konfigurasi berikut

TX 3DR                      RX Kontroller(PA\_12)

RX 3DR TX Kontroller(PA\_11)  
 VCC 5Volt  
 GND GND

Setelah itu pada program diberikan perintah untuk dabar mengirim data ke HMI dengan dengan perintah `pc.readable( )` sebagai default agar HMI dapat memaca data dan `pc.printf( )` agar dapat mecetak string dari kontroler.



**Gambar 3. 3 3DR Config**

#### D. SD CARD

SD card berfungsi sebagai penyimpan data logger pembacaan arus dan tegangan dan disimpan dalam format .txt. Konfigurasi pin yang dirancang adalah MISO, MOSI, SCK, SDCS dengan pin pada controller D11, D12, D13, D10. Untuk VCC SD card disarankan menggunakan 3,3 Volt agar tidak merusak SD Card. SD CARD menggunakan format FAT32 dan memiliki kapasitas memori 16GB.

```

64 void sdcard()
65 {
66     printf("SD thread is ready\r\n");
67     mkdir("/sd/Progress2", 0777);
68
69     FILE *fp = fopen("/sd/Progress2/sdtest.txt", "a+");
70     if(fp == NULL) {
71         error("Could not open file for write\n");
72     }
73     fprintf(fp, "Arus=%f\t Tegangan=%f\t Daya=%f\t\n", Arus ,Tegangan, Daya);
74     fclose(fp);
75 }

```

**Gambar 3. 4** Pseudocode SD Card

Dari program SD card pada gambar 3.4 pada baris 67 terdapat angka 0777 yang berarti sebuah hak akses untuk user agar dapat me read write data kemudian di save dalam bentuk .txt dalam folder yang bernama Progress2 pada baris 67 yang juga merupakan hasil perintah dari syntax mkdir atau make directory. Selain itu Pin pada module slot SD card harus tersambung secara benar terhadap kontroler yaitu sebagai berikut

MISO SDCARD	MISO KONTROLLER(11)
MOSI SDCARD	MOSI KONTROLLER(12)
SDCS SDCARD	SDCS KONTROLLER(10)
SCK SDCARD	SCK KONTROLLER(13)

### 3.2.2 Perancangan Software

Pada poin ini terdapat beberapa poin terkait perancangan software yaitu sebagai berikut:

#### A.HMI

HMI atau Human Machine Interface berfungsi sebagai perantara antara User dan Mesin disini HMI dibuat berupa aplikasi monitoring yang dapat menerima dan membaca data arus dan tegangan serta daya dari system tracking fotovoltaic dan menampilkannya pada PC. Untuk pembuatan aplikasi dilakukan pada Software Visual Studio menggunakan Bahasa pemrograman C#

```

1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using System.ComponentModel;
5  using System.Data;
6  using System.Drawing;
7  using System.Linq;
8  using System.Linq.Expressions;
9  using System.Text;
10 using System.Threading.Tasks;
11 using System.Windows.Forms;
12 using System.IO.Ports;
13 using System.IO;
14 using System.Threading;
15 using System.Diagnostics;
16 using rtChart;
17 using System.Windows.Forms.DataVisualization.Charting;
18 using System.Globalization;

```

**Gambar 3. 5 Header C#**

Header berfungsi memanggil library untuk header tambahan yaitu Threading dan diagnostic serta rtchart yang berfungsi untuk mengupdate grafik. Sedangkan IO Port merupakan definisi input atau output untuk mengaktifkan koneksi antar port. pada gambar 3.5 ditunjukkan apasaja header yang digunakan.

```

22 public partial class Form1 : Form
23 {
24
25     private SerialPort myPort;
26     private DateTime dateTime;
27     private string data_in;
28     private string data_in_div;
29     private string data_end;
30     int data_ln,x_i, a=0, b=0;
31     kayChart ch_data;
32
33     public string amper_s, daya_s, voltages_s, Error_s;
34     double amper, daya, voltages;
35     double amper_, daya_, voltages_, Error;
36     NumberStyles styles;
37
38     double conv_result;
39

```

**Gambar 3. 6 Pseudocode variable yang akan ditampilkan pada Aplikasi**

Selanjutnya pada gambar 3.6 pendeklarasian variable-variabel yang akan digunakan pada program ini, seperti pada baris 25 merupakan pendeklarasian varial untuk port penerima, pada baris 33 merupakan deklarasi variable untuk umum atau dapat digunakan pada tiap fungsi di dalam form yaitu amper\_s untuk arus, daya\_s untuk daya dan Voltages\_s sebagai tegangan dan tipe datanya apakah string untuk huruf dan double untuk angka decimal.

```

45 private void StartButton_Click(object sender, EventArgs e)
46 {
47     myPort = new SerialPort();
48
49     if (textBox_portname.Text == "")
50     {
51         MessageBox.Show("PORT NAME is EMPTY");
52         textBox_portname.Focus();
53     }
54     if (textBox_baud.Text == "")
55     {
56         MessageBox.Show("BAUDRATE is EMPTY");
57         textBox_baud.Focus();
58     }
59     int baud = Convert.ToInt32(textBox_baud.Text);
60     myPort.PortName = textBox_portname.Text; //textBox_portname.Text;
61     myPort.BaudRate = baud; //57600;
62     myPort.Parity = Parity.None;
63     myPort.StopBits = StopBits.One;
64     myPort.DataBits = 8;
65     myPort.DataReceived += myPort_DataReceived;

```

**Gambar 3. 7** Pseudocode Start button

Dari variable yang telah di tentukan selanjutnya membuat perintah ketika tombol start pada HMI ditekan, yaitu akan memberi perintah pada textbox untuk menerima data yang diterima dari port serial tetapi harus dipastikan bahwa textbox portname dan baud sudah terisi dengan benar. baud rate default diatur yaitu sebesar 57600 seperti yang tertera pada gambar 3.7 baris 61.

```

67         try
68         {
69             textBox1.Text = "";
70             myPort.Open();
71             if (data_in_div == "A")
72             {
73                 textBox_datatx_arus.Text = "";
74             }
75             if (data_in_div == "P")
76             {
77                 textBox_datatx_daya.Text = "";
78             }
79             if (data_in_div == "V")
80             {
81                 textBox_datatx_bayess.Text = "";
82             }
83         }
84         catch (Exception ex)
85         {
86             MessageBox.Show(ex.Message, "Error");
87         }
88     }

```

**Gambar 3. 8** Pseudocode Serial Port

Pada gambar 3.8 Ketika start telah ditekan maka serial port akan mulai menerima data dan akan menuliskan pada textbox yang telah ditentukan yaitu arus, daya dan tegangan. Karena terdapat tiga data yang masuk dan diterima secara bersamaan maka dibuat program pemisah yaitu dengan berdasarkan inisialisasi pada hasil penerimaan di serial port. Dengan ketentuan apabila didepan nilai yang diterima terdapat huruf "A" maka akan dipisah dan dituliskan pada textbox hasil pengukuran Arus. Apabila didepan nilai terdapat huruf "P" berarti nilai tersebut merupakan hasil pengukuran daya dan dimasukan textbox daya. Yang terakhir apabila data yang diterima terdapat huruf "V" maka nilai tersebut akan dimasukan ke dalam textbox hasil pengukuran tegangan.

```

226 private void displaydata_event(object sender, EventArgs e)
227 {
228     DateTime date = DateTime.Now;
229     time = date.Day + "/" + date.Month + "/" + date.Year + ":" + date.Hour + ":" + date.Minute + ":" + date.Second;
230     styles = NumberStyles.AllowTrailingSign | NumberStyles.AllowThousands;
231     textBox1.AppendText("\t"+time+"\n");
232
233     if (data_in_div == "A")
234     {
235
236         amper_s = data_end;
237         textBox_datatx_arus.AppendText("\t"+amper_s+"\n");
238         chart_AP.Series[0].Points.AddXY(0, amper_s);
239     }
240     if (data_in_div == "P")
241     {
242         daya_s = data_end;
243         textBox_datatx_daya.AppendText("\t"+daya_s+"\n");
244         chart_AP.Series[1].Points.AddXY(0, daya_s);
245     }
246     if (data_in_div == "V")
247     {
248         voltages_s = data_end;
249         textBox_datatx_bayess.AppendText("\t" + voltages_s + "\n");
250         chart_AP.Series[2].Points.AddXY(0, voltages_s);
251     }
252 }

```

**Gambar 3. 9** Pseudocode Grafik Update

Data yang diperoleh dari pembacaan serial port tadi juga akan ditampilkan dalam bentuk grafik, yaitu dengan cara memperbarui nilai chart yaitu chart nol untuk data arus chart 1 untuk data daya dan chart 2 untuk data tegangan Dapat dilihat pada gambar 3.9

```

90 private void circullarButton1_Click(object sender, EventArgs e)
91 {
92     timer1.Enabled = false;
93     try
94     {
95         myPort.Close();
96     }
97     catch (Exception ex2)
98     {
99         MessageBox.Show(ex2.Message, "Error");
100     }
101 }

```

**Gambar 3. 10** Pseudocode Stop Button

ketika ingin memberhentikan monitoring data ,maka cukup tekan tombol stop perintah tersebut terdapat pada baris 90 di gambar 3.10 dan program akan menjalankan perintah untuk menutup komunikasi dengan port serial maka algoritma akan memberi

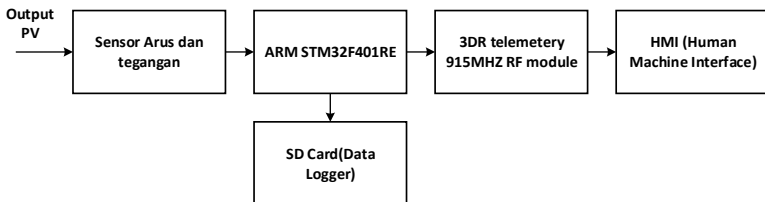
perintah untuk menutup MyPort atau komunikasi dengan port dengan perintah seperti pada baris 95



## BAB IV HASIL DAN PEMBAHASAN

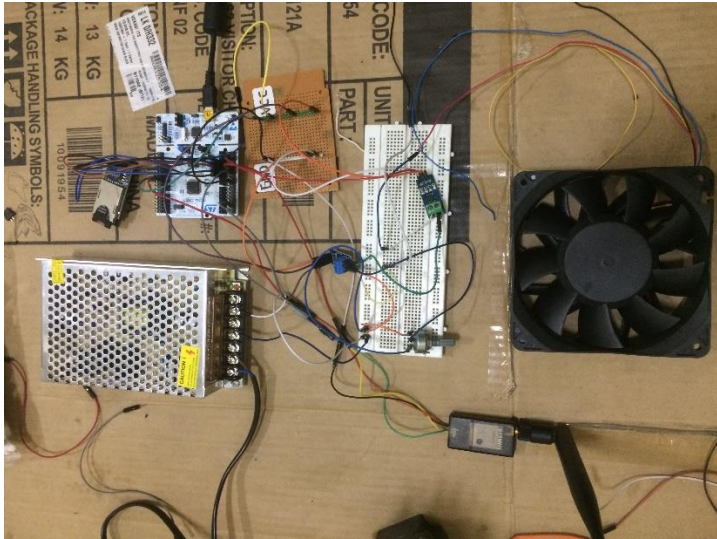
### 4.1 Hasil Pengerjaan Sistem

Sistem monitoring adalah sebuah sistem untuk mengamati hasil perubahan suatu besaran fisis pada objek atau benda yang ingin kita ukur, serta dapat di informasikan berupa tampilan atau biasa disebut HMI(Human Machine Interfacing). Sehingga pengguna dapat mengamati perubahan besaran fisis dengan mudah secara real time. Berikut ini adalah diagram blok monitoring arus tegangan dan daya pada sistem *tracking* fotovoltaik yang ditunjukkan oleh gambar 4.1

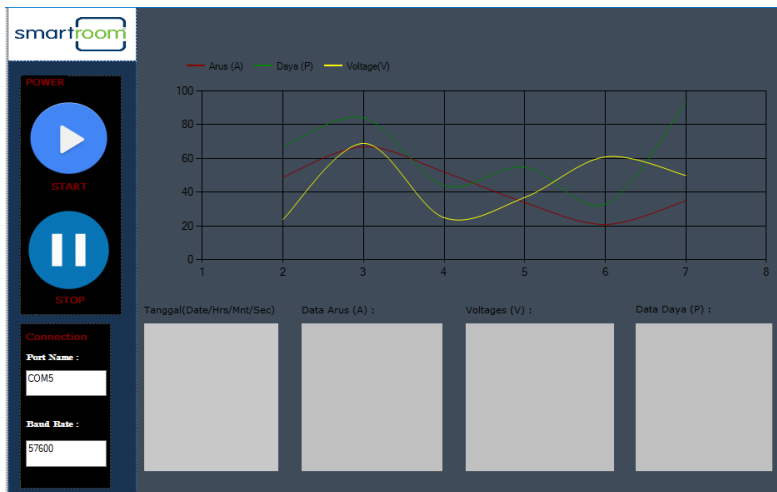


**Gambar 4. 1** Diagram Blok Sistem Monitoring Tracking Fotovoltaik

Berdasarkan Gambar 4.1 terdapat berbagai komponen monitoring pada sistem tracking fotovoltaic untuk memonitoring nilai arus, tegangan dan daya hasil keluaran dari fotovoltaic atau dari sumber tegangan dan arus yang lainnya, nilai ini harus akurat dan real time agar data yang didapat dapat menjadi acuan apabila ingin diolah lebih lanjut dan dikirim melalui komunikasi data berbasis wireless. Telah di realisasikan hardware sesuai dengan diagram blok diatas untuk mengetahui apakah system monitoring tracking berbasis wireless yang dirancang telah memenuhi harapan atau belum.



**Gambar 4. 2** Hasil Perancangan Hardware



**Gambar 4. 3** Hasil Perancangan Software

Berdasarkan gambar 4.2 telah berhasil dirancang hardware berupa sensor tegangan dan arus untuk mengukur besaran fisis kemudian supply 24 VDC sebagai Vcc atau supply dengan beban

sebuah fan dengan daya 35 Watt dengan input minimum 24VDC dan arus 1 Ampere. Gambar 4.3 merupakan hasil perancangan aplikasi berupa software.exe yang dapat dibuka di platform windows, mac, atau linux. Aplikasi ini dapat mengukur arus dan tegangan yang kemudian dikalikan dan menghasilkan Daya dengan penunjuk waktu agar mempermudah pengolahan data. Dalam perancangan software aplikasi pada gambar 4.3 koding yang dirancang adalah sebagai berikut.

---

```
#include "mbed.h"
#include "rtos.h"
#include "SDFileSystem.h"
#define E1 2.96

Thread thread;

AnalogIn potentio(PA_0);
DigitalOut led(LED1);
AnalogIn analog_Arus(PC_5);
AnalogIn analog_Tegangan (PC_4);

//PwmOut mypwm(PC_6);

Serial pc(PA_11, PA_12);

SDFileSystem sd(D11, D12, D13, D10, "sd");

float Arus, Tegangan, Arus_ADC, Tegangan_ADC, del_v, Daya;
//float LDR1, LDR2, LDR1_temp, LDR2_temp, LDR1_aver,
LDR2_aver;

void sdcard();
void getADC_LDR();

void thread_AVP()
{
```

```

    while (true) {
        wait(0.3);
        Tegangan_ADC = analog_Tegangan.read();
        Arus_ADC = analog_Arus.read();
        Tegangan = (((Tegangan_ADC/5)*1024)/10)+E1;
        del_v = ((Arus_ADC/5)*1024)-2.5;
        Arus = del_v/100 ;
        pc.printf("V%.2f\n", Tegangan);
        pc.printf("A%.2f\n", Arus);
        Daya=Tegangan*Arus;
        pc.printf("P%.2f\n", Daya);
        sdcard();
    }
}

```

```

int main()
{
    //printf("\n\n*** RTOS basic example ***\n");

```

```

pc.readable();
pc.baud(57600);

```

```

thread.start(thread_AVP);

```

```

//thread.start(thread_pwm);
//thread.start(thread_LDR);

```

```

    while (true) {
        wait(0.3);
        led = 1;
    }
}

```

```

void sdcard()
{
    printf("SD thread is ready\r\n");
}

```

```

mkdir("/sd/Progress2", 0777);

FILE *fp = fopen("/sd/Progress2/sdtest.txt", "a+");
if(fp == NULL) {
    error("Could not open file for write\n");
}
fprintf(fp, "Arus=%f\t Tegangan=%f\t Daya=%f\t\n", Arus
,Tegangan, Daya);
    fclose(fp);
}

```

---

Dari koding diatas akan dihasilkan pembacaan Arus dan tegangan yang akan di kirim ke HMI monitoring system Tracking melalui module wireless 3dr Telemetry dengan data arus yang dikirim diawali dengan A, tegangan diawali dengan V, dan daya diawali dengan P. Baud rate yang digunakan adalah 57600 yang merupakan default dari module 3DR 915 MHZ kemudian data pembacaan arus dan tegangan serta daya akan disimpan ke dalam SD Card selama mikro menyala hingga mikro mati.

---

```

using System;
using System.Collections;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Linq.Expressions;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO.Ports;
using System.Threading;
using System.Diagnostics;
using rtChart;
using System.Windows.Forms.DataVisualization.Charting;
using System.Globalization;

namespace COM_Form
{

```

```

public partial class Form1 : Form
{
    private SerialPort myPort;
    private DateTime dateTime;
    private string data_in;
    private string data_in_div;
    private string data_end;
    int data_ln,x_i, a=0, b=0;
    kayChart ch_data;

    private string amper_s, daya_s, voltages_s, Error_s;
    double amper, daya, voltages;
    double amper_, daya_, voltages_, Error;
    NumberStyles styles;

    double conv_result;

    private void textBox_portname_TextChanged(object
sender, EventArgs e)
    {

    }

    private void StartButton_Click(object sender, EventArgs
e)
    {
        myPort = new SerialPort();

        if (textBox_portname.Text == "")
        {
            MessageBox.Show("PORT NAME is EMPTY");
            textBox_portname.Focus();
        }
        if (textBox_baud.Text == "")
        {
            MessageBox.Show("BAUDRATE is EMPTY");
            textBox_baud.Focus();
        }
    }
    int baud = Convert.ToInt32(textBox_baud.Text);

    myPort.PortName = textBox_portname.Text;
    //textBox_portname.Text;

```

```

myPort.BaudRate = baud; //57600;
myPort.Parity = Parity.None;
myPort.StopBits = StopBits.One;
myPort.DataBits = 8;
myPort.DataReceived += myPort_DataReceived;

try
{
    textBox1.Text = "";
    myPort.Open();
    if (data_in_div == "A")
    {
        textBox_datatx_arus.Text = "";
    }
    if (data_in_div == "P")
    {
        textBox_datatx_daya.Text = "";
    }
    if (data_in_div == "V")
    {
        textBox_datatx_bayess.Text = "";
    }
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "Error");
}

private void circularButton1_Click(object sender,
EventArgs e)
{
    try
    {
        myPort.Close();
    }
    catch (Exception ex2)
    {
        MessageBox.Show(ex2.Message, "Error");
    }
}

```

```
private void chart_AP_Click(object sender, EventArgs
e)
{

}

private void label_data_daya_Click(object sender,
EventArgs e)
{

}

private void textBox1_TextChanged(object sender,
EventArgs e)
{

}
private void textBox1_TextChanged_1(object sender,
EventArgs e)
{

}

private void Form1_Load_1(object sender, EventArgs e)
{

}

private void textBox1_TextChanged_2(object sender,
EventArgs e)
{

}

private void label3_Click(object sender, EventArgs e)
{

}

private void pictureBox1_Click(object sender, EventArgs
e)
{
```



```

}

private void panel1_Paint(object sender, PaintEventArgs
e)
{

}

private void panel2_Paint(object sender, PaintEventArgs
e)
{

}

private void txtCB_TextChanged(object sender, EventArgs
e)
{

}

double xValues =0;

Expression abc;

public Form1()
{
    InitializeComponent();
}

Konversi ubah = new Konversi();

private void Form1_Load(object sender, EventArgs e)
{
    ch_data = new kayChart(chart_AP, 60);
    ChartArea area = chart_AP.ChartAreas[1];

    area.AxisX.Minimum = 1;
    area.AxisX.Maximum = 10;
    area.AxisY.Minimum = 1;
    area.AxisY.Maximum = 10;
}

```

```

        //ch_data.serieName = "Arus (A)";
    }

private void button_start_Click(object sender,
EventArgs e)
{

}

void myPort_DataReceived(object sender,
SerialDataReceivedEventArgs e)
{
    data_in = myPort.ReadLine();
    data_ln = data_in.Length;
    data_end = data_in.Substring(1, data_ln-1);
    data_in_div = data_in.Substring(0, 1);
    this.Invoke(new EventHandler(displaydata_event));
}

private void displaydata_event(object sender, EventArgs
e)
{
    dateTime = DateTime.Now;
    styles = NumberStyles.AllowTrailingSign |
    NumberStyles.AllowThousands;
    //double[] amper_ch = new Double[10];
    //list_amp = new List<double>();
    //ArrayList list_amp = new ArrayList();
    string time = dateTime.Day + "/" + dateTime.Hour + ":"
    + dateTime.Minute + ":" + dateTime.Second;
    textBox1.AppendText("\t"+time+"\n");

    if (data_in_div == "A")
    {
        //textBox_datatx_arus.AppendText(time + "\t\t\t" +
        data_end + "\n");
        //amper = Double.Parse(data_end);
        //amper_s = amper.ToString("G",
        CultureInfo.InvariantCulture);
        //amper_ch[a] = amper;
        amper_s = data_end;
    }
}

```

```

textBox_datatx_arus.AppendText("\t"+amper_s+"\n");
//amper_ch = amper;
//list_amp.Add(amper_ch[a]);
chart_AP.Series[0].Points.AddXY(0, amper_s);
//a++;
}
if (data_in_div == "P")
{
//textBox_datatx_daya.AppendText(time + "\t\t\t" +
data_end + "\n");
//daya = Double.Parse(data_end);
//daya_ = daya/100;
daya_s = data_end;
daya_s = daya_s;

textBox_datatx_daya.AppendText("\t"+daya_s+"\n");
chart_AP.Series[1].Points.AddXY(0, daya_s);
}
if (data_in_div == "V")
{
//bayess_ = bayess;
voltages_s = data_end / 10;
//voltages_s /= 10;
textBox_datatx_bayess.AppendText("\t" + voltages_s +
"\n");
chart_AP.Series[2].Points.AddXY(0, voltages_s);
//bayess = ubah.ToDecimal(bayess_s);
//txtCB.AppendText(bayess+"\n");
//txtCB.Text = Convert.ToString(bayess);

//.....Error

//Error_s = string.Format("");
//textBox_error.AppendText(" " + Error_s + "\n");
}
}

private void textBox_datatx_arus_TextChanged(object
sender, EventArgs e)
{

```

```

textBox_datatx_arus.SelectionStart =
textBox_datatx_arus.Text.Length;
textBox_datatx_arus.ScrollToCaret();
}

private void textBox_datatx_daya_TextChanged(object
sender, EventArgs e)
{
textBox_datatx_daya.SelectionStart =
textBox_datatx_daya.Text.Length;
textBox_datatx_daya.ScrollToCaret();
}

}
}

```

---

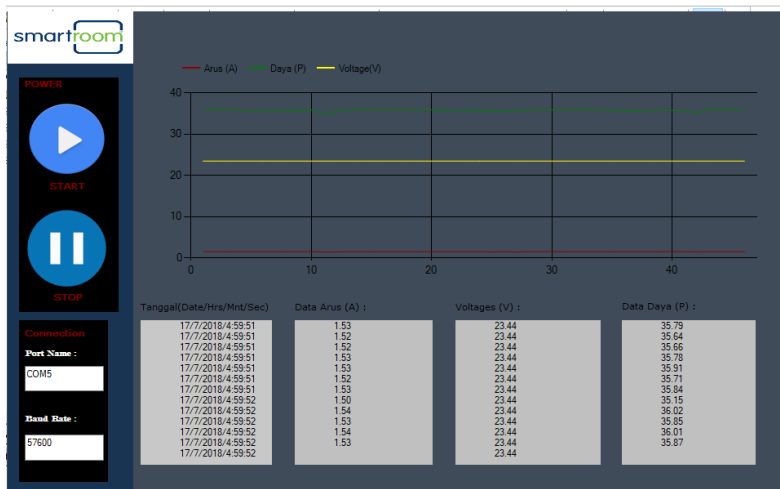
Dari program diatas didapatkan Aplikasi untuk memonitoring Arus, tegangan, dan daya berbasis wireless dengan modul 3dr telemetry secara real time. Selain itu data dari mikro akan ditampilkan dalam grafik yang juga akan mengupdate data secara realtime.

---

#### 4.1.1 Pengujian Alat Dengan Data Pada SD Card

Pengujian alat dilakukan dengan merancang hardware untuk menghitung besaran fisis berupa daya, arus, dan tegangan dengan memanipulasi sumber dengan potensiometer. Sumber daya yaitu power suplai 24VDC dengan beban sebuah kipas dengan daya 35.5 watt dan input 24 -36 VDC dan diamati respon dari HMI, agar HMI dapat mulai membaca data maka perlu ditekan tombol start pada interface. Ketika tombol start sudah ditekan textbox akan menerima data dan akan mengupgrade grafik sehingga nilai grafik sama seperti data pada textbox. Untuk data logger akan tersimpan otomatis pada sdcard dengan format (dot).txt. Aplikasi ini memiliki delay untuk pembacaannya karena data serial mengirimkan satu persatu bit data, dan ketika di running timer akan berjalan dahulu karena timer berasal dari aplikasi bukan data yang dikirim dari Mikrokontroler. berikut

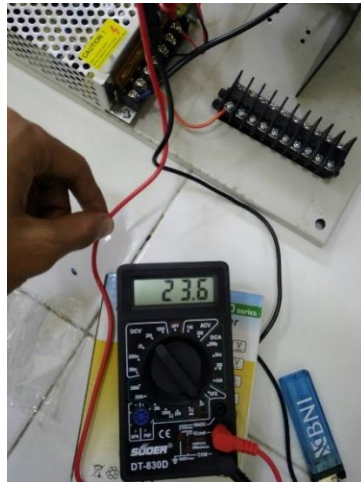
data pembacaan Arus, Daya, dan Tegangan pada aplikasi HMI dan file.txt pada SD Card pada gambar 4.4 dan gambar 4.5.



**Gambar 4. 4** Pembacaan Data Pada HMI

Arus=1.530380	Tegangan=23.439999	Daya=35.872101
Arus=1.527379	Tegangan=23.439999	Daya=35.801765
Arus=1.530380	Tegangan=23.439999	Daya=35.872101
Arus=1.533381	Tegangan=23.439999	Daya=35.942440
Arus=1.534381	Tegangan=23.439999	Daya=35.965885
Arus=1.527379	Tegangan=23.439999	Daya=35.801765
Arus=1.526879	Tegangan=23.439999	Daya=35.790039
Arus=1.520378	Tegangan=23.439999	Daya=35.637646
Arus=1.521378	Tegangan=23.439999	Daya=35.661091
Arus=1.526379	Tegangan=23.439999	Daya=35.778320
Arus=1.531880	Tegangan=23.439999	Daya=35.907269
Arus=1.523378	Tegangan=23.439999	Daya=35.707981
Arus=1.528880	Tegangan=23.439999	Daya=35.836933
Arus=1.499372	Tegangan=23.439999	Daya=35.145283
Arus=1.536881	Tegangan=23.439999	Daya=36.024498
Arus=1.529380	Tegangan=23.439999	Daya=35.848656
Arus=1.536381	Tegangan=23.439999	Daya=36.012775
Arus=1.530380	Tegangan=23.439999	Daya=35.872101
Arus=1.529880	Tegangan=23.439999	Daya=35.860382
Arus=1.534881	Tegangan=23.439999	Daya=35.977608
Arus=1.522378	Tegangan=23.439999	Daya=35.684536
Arus=1.524879	Tegangan=23.439999	Daya=35.743149
Arus=1.531380	Tegangan=23.439999	Daya=35.895550
Arus=1.531380	Tegangan=23.439999	Daya=35.895550
Arus=1.523878	Tegangan=23.439999	Daya=35.719707

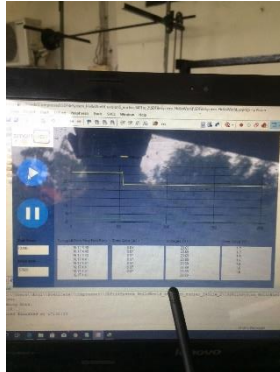
**Gambar 4. 5** Pembacaan Data SD Card



**Gambar 4. 6** Pembacaan Sumber Tegangan oleh Multimeter

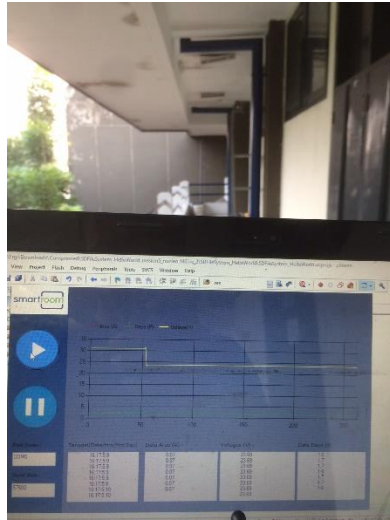
#### 4.1.2 Pengujian Alat Dengan Variasi Jarak

Pada pengujian alat kali ini dilakukan dengan memanipulasi jarak antara PC dengan Mikro untuk mengetahui jarak maksimum pengiriman data menggunakan modul wireless 3DR Telemetry.



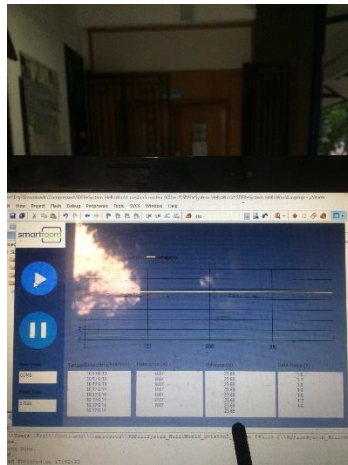
**Gambar 4. 7** Pembacaan Data Dengan Jarak 1 Meter

Untuk pembacaan dengan jarak 1 meter penulis berjalan keluar ruang A300. Data masih dapat terbaca dengan lancar dengan kecepatan yang tetap.



**Gambar 4. 8** Pembacaan Dengan Jarak 5 Meter

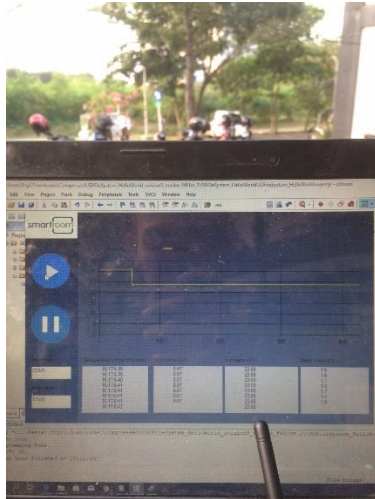
Dengan jarak 5 meter data dapat terbaca dengan lancar dengan kecepatan *transfer* yang sama. Pengujian dengan jarak kurang lebih 5 meter dihitung dari transmitter yang berada di ruang A300 dan receiver berada di depan kelas A302



**Gambar 4. 9** Pembacaan Dengan Jarak 15 Meter

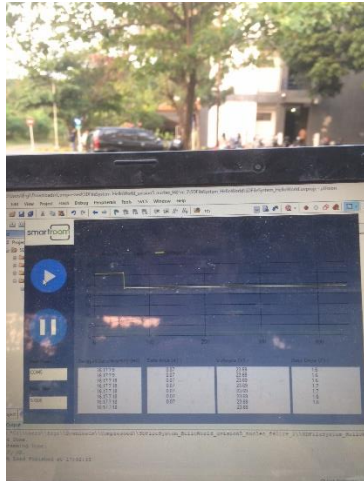


Dengan jarak 15 meter aplikasi monitoring pun juga mampu menerima data dengan lancar walau terhalang bangunan gedung teknik instrumentasi yang berbahan beton



**Gambar 4. 10** Pembacaan Dengan Jarak 25 Meter

Kemudian dengan jarak sejauh 25 meter, data masih tetap dapat terbaca dengan kecepatan yang sama. Dengan sumber transmitter di ruang A300 dan Laptop di depan tempat parkir mobil teknik instrumentasi



**Gambar 4. 11** Pembacaan Dengan Jarak 50 Meter

Dengan jarak sejauh 50 meter, pembacaan data sudah mulai ada *delay* pembacaan data namun masih dapat menerima dan membaca data.



**Gambar 4. 12** Pembacaan Dengan Jarak 100 Meter

Dengan jarak 100 Meter aplikasi monitoring tidak dapat menerima data dari mikro. Modul 3DR memberikan petunjuk lampu hijau berkedip dan tidak ada tanda merah yang berarti

tidak tersambung dengan transmitter dan tidak sedang menerima data, yang bilamana dia tersambung maka led indicator hijau akan menyala tidak berkedip dan bila terjadi komunikasi data maka led indicator warna merah akan berkedip.

## 4.2 Pembahasan

Dari data yang didapat, diketahui bahwa software aplikasi HMI dapat menerima data secara real time dan dapat memisahkan data arus, daya, dan tegangan secara bersamaan. Dengan data yang dibuat berubah-ubah berdampak pada bentuk grafik yang naik dan turun mengikuti nilai data. Nilai daya di dapat dari mengkalikan hasil pembacaan sensor tegangan dan arus. Sehingga pada grafik nilai sensor tegangan selalu berada di atas pembacaan data lainnya. Data yang kedua didapatkan dari SD Card yang berfungsi sebagai data logger yang bertujuan untuk pengarsipan data dari pengukuran sensor. Dari hasil uji coba alat didapatkan range pembacaan Tegangan sebesar 23.44 VDC dan arus sebesar 1.5 Ampere. Bila dibandingkan dengan pembacaan multimeter terdapat perbedaan pembacaan sebesar 0.2 yang berarti alat ini masih mampu membaca secara akurat karena toleransi kesalahan pembacaan adalah 2%. Kemudian untuk arus, diketahui daya kipas adalah 3.55 watt maka arus dapat dihitung dengan rumus  $\text{Arus} = \text{Daya} / \text{Tegangan}$ , dan didapatkan hasil sebesar 1.5. Terdapat perbedaan pembacaan karena pada HMI pembacaan data dimulai sesuai kehendak pengguna untuk memulai atau memberhentikan, tetapi berbeda pada SD Card yaitu ketika sistem pengukuran berjalan maka data hasil pembacaan sensor secara otomatis akan tersimpan sebagai logger atau riwayat hingga sistem mati atau SD Card di lepas. Selain itu terjadi perbedaan banyak jumlah dibelakang koma sehingga di SD Card terbaca 23.399 yang bila dibulatkan menjadi sama seperti pada aplikasi HMI yaitu 23.44. Kemudian untuk pengujian data dengan memanipulasi jarak, didapatkan hasil bahwa jarak terjauh pembacaan modul wireless 3dr Telemetry kurang lebih sejauh 100 Meter. Sedangkan pada *datasheet* tertulis bahwa bila tidak ada gangguan(*noise*) maka modul 3dr dapat berkomunikasi sejauh 300 atau bisa lebih. Selain itu karena data yang dikirimkan sejumlah 3 variabel sehingga dapat

mengurangi kecepatan pengiriman data dan semakin berkurangnya jarak terjauh pengiriman atau komunikasi data.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan rumusan masalah pada BAB I didapatkan kesimpulan sebagai berikut

1. Dari hasil pembacaan data arus, daya , dan tegangan pada aplikasi monitoring system tracking, didapatkan bahwa data yang ditampilkan oleh aplikasi dapat diterima secara realtime dan disimpan sebagai logger pada SD Card.
2. Aplikasi dapat memanfaatkan komunikasi data berbasis wireless menggunakan modul 3dr telemetry dengan baik, dan dapat menjangkau jarak yang diinginkan.

#### **5.2 Saran**

Berdasarkan penelitian yang telah dilakukan terdapat beberapa saran diantaranya sebagai berikut :

1. Perlu dilakukan pengembangan kedepannya, agar sistem HMI dapat untuk mengontrol aktuator pada plant.

(Halaman ini sengaja dikosongkan)

## DAFTAR PUSTAKA

- [1] K. Akhmad, Pembangkit Listrik Tenaga Surya dan Penerapannya Untuk Daerah Terpencil, 2011.
- [2] A. K. S. d. A. F. Rahayuningtyas, “ Studi Perencanaan Sistem Pembangkit Listrik Tenaga Surya (Plts) Skala Rumah Sederhana Di Daerah Pedesaan Sebagai Pembangkit Listrik Alternatif Untuk Mendukung Program Ramah Lingkungan Dan Energi Terbarukan,” *Prosiding SnaPP 2014 Sains, Teknologi, dan Kesehatan*, pp. 223-230, 2014.
- [3] A. T. Soelistio, “SOLAR POWER PLANTS,” *PAPER PEMBANGKIT LISTRIK TENAGA SURYA (PLTS)*.
- [4] Kahfi, A., Firdaus & Mohammad, H. H, “Rancang Bangun Sistem Kendali Tracking Sinar Matahari Dengan Logika Fuzzy,” *SEMANTIK*, 2015.
- [5] Qing Liu, Dikpride Despa, YasUNOri Mitani, “Application of Phasor and Node Voltage Measurements to Monitoring Power Flow Stability,” *International Journal on Electrical Engineering and Informatics*, vol. Vol. 4, 2012.
- [6] Dikpride Despa, Ady Kurniawan, M. Komarudin, Mardiana, Gigih F N, “Smart Monitoring of Electrical Quantities Based on Single Board Computer BCM2835,” 2015.
- [7] K. A. e. sut, “FPGA BASED MAXIMUM POWER POINT TRACKING OF PHOTOVOLTAIC SYSTEM,” 2014.
- [8] “RANCANG BANGUN SOLAR TRACKER BERBASIS MIKROKONTROLER ATmega8535 DENGAN SENSOR LDR DAN PENAMPIL LCD,” *Jurnal Fisika Unand*, vol. vol 2 no 4, 2016.
- [9] Girsang, Irma Sika, “Perancangan monitorng jarak jauh ketinggian air pada bendungan menggunakan sistem android via Wi-Fi,” *skripsi USU*, 2014.
- [10] Heri Haryanto, Sarif Hidayat, “Perancangan HMI (Human Machine Interface),” *SETRUM*, Vol. %1 dari %2Volume 1, No. 2, 2012.
- [11] S. S, “Teknik Pengukuran Listrik dan Elektronika,” 1999.

- [12] HILMAN HR. JUFRI, NASRUDDIN M.N, BISMAN P, “RANCANG BANGUN ALAT UKUR DAYA ARUS BOLAK-BALK BERBASIS MIKROKONTROLER ATMEGA8535”.
- [13] st.com, “stm32f401re,” ARM, [Online]. Available: <https://www.st.com/en/microcontrollers/stm32f401re.html>. [Diakses sabtu juli 2018].
- [14] RATNA SUSANA, MUHAMMAD ICHWAN, SAVERO AL PHARD, “Penerapan Metoda Serial Peripheral Interface(SPI) pada rancang bangun data logger berbasis SD Card,” *Jurnal ELKOMIKA*, vol. 4, pp. 208-227, 2016.
- [15] 3. Robotics, “3DR RADIO V2 QUICK START GUIDE,” 2013.
- [16] A. MBED, “MBED HOME,” ARM MBED, 2018. [Online]. Available: <https://www.mbed.com/en/>. [Diakses sabtu juli 2018].



# LAMPIRAN

## Lampiran A Datasheet STM32F401RE



life.augmented

### STM32F401xD STM32F401xE

ARM® Cortex®-M4 32b MCU+FPU, 105 DMIPS,  
512KB Flash/96KB RAM, 11 TIMs, 1 ADC, 11 comm. interfaces

Datasheet - production data

#### Features

- Core: ARM® 32-bit Cortex®-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 84 MHz, memory protection unit, 105 DMIPS/1.25 DMIPS/MHz (Dhrystone 2.1), and DSP Instructions
- Memories
  - up to 512 Kbytes of Flash memory
  - up to 96 Kbytes of SRAM
- Clock, reset and supply management
  - 1.7 V to 3.6 V application supply and I/Os
  - POR, PDR, PVD and BOR
  - 4-to-26 MHz crystal oscillator
  - Internal 16 MHz factory-trimmed RC
  - 32 kHz oscillator for RTC with calibration
  - Internal 32 kHz RC with calibration
- Power consumption
  - Run: 146 µA/MHz (peripheral off)
  - Stop (Flash in Stop mode, fast wakeup time): 42 µA Typ @ 25°C; 65 µA max @ 25 °C
  - Stop (Flash in Deep power down mode, fast wakeup time): down to 10 µA @ 25 °C; 30 µA max @ 25 °C
  - Standby: 2.4 µA @ 25 °C / 1.7 V without RTC; 12 µA @ 85 °C @ 1.7 V
  - V<sub>BAT</sub> supply for RTC: 1 µA @ 25 °C
- 1x12-bit, 2.4 MSPS A/D converter: up to 16 channels
- General-purpose DMA: 16-stream DMA controllers with FIFOs and burst support
- Up to 11 timers: up to six 16-bit, two 32-bit timers up to 84 MHz, each with up to four IC/OC/PWM or pulse counter and quadrature (Incremental) encoder input, two watchdog timers (independent and window) and a SysTick timer



- Debug mode
  - Serial wire debug (SWD) & JTAG interfaces
  - Cortex®-M4 Embedded Trace Macrocell™
- Up to 81 I/O ports with interrupt capability
  - Up to 78 fast I/Os up to 42 MHz
  - All I/O ports are 5 V-tolerant
- Up to 12 communication interfaces
  - Up to 3 x I<sup>2</sup>C interfaces (SMBus/PMBus)
  - Up to 3 USARTs (2 x 10.5 Mbit/s, 1 x 5.25 Mbit/s), ISO 7816 Interface, LIN, IrDA, modem control)
  - Up to 4 SPIs (up to 42 Mbit/s at f<sub>CLK</sub> = 84 MHz), SPI2 and SPI3 with muxed full-duplex I<sup>2</sup>S to achieve audio class accuracy via internal audio PLL or external clock
  - SDIO Interface
  - Advanced connectivity: USB 2.0 full-speed device/host/OTG controller with on-chip PHY
- CRC calculation unit
- 96-bit unique ID
- RTC: subsecond accuracy, hardware calendar
- All packages (WLQFP49, LQFP100, UQFPN48, UFBGA100) are ECOPACK®2

Table 1. Device summary

Reference	Part number
STM32F401xD	STM32F401CD, STM32F401RD, STM32F401VD
STM32F401xE	STM32F401CE, STM32F401RE, STM32F401VE

## Lampiran B Datasheet Sensor ACS712

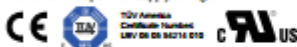


ACS712

*Fully Integrated, Hall Effect-Based Linear Current Sensor  
with 2.1 kVRMS Voltage Isolation and a Low-Resistance Current Conductor*

### Features and Benefits

- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5  $\mu$ s output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at  $T_A = 25^\circ\text{C}$
- Small footprint, low-profile SOIC8 package
- 1.2 m $\Omega$  internal conductor resistance
- 2.1 kV<sub>ISO</sub> minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Factory-trimmed for accuracy
- Extremely stable output offset voltage
- Nearly zero magnetic hysteresis
- Ratio-metric output from supply voltage



Package: 8 Lead SOIC (suffix LC)



Approximate Scale 1:1

### Description

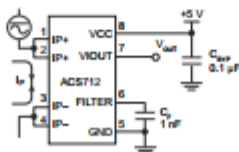
The Allegro® ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems. The device package allows for easy implementation by the customer. Typical applications include motor control, load detection and management, switched-mode power supplies, and overcurrent fault protection.

The device consists of a precise, low-offset, linear Hall sensor circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which is sensed by the integrated Hall IC and converted into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging.

The output of the device has a positive slope ( $>V_{OUT(0)}$ ) when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sensing. The internal resistance of this conductive path is 1.2 m $\Omega$  typical, providing low power

*Continued on the next page...*

### Typical Application



Application 1. The ACS712 outputs an analog signal,  $V_{OUT}$ , that varies linearly with the uni- or bi-directional AC or DC primary sensed current,  $I_p$ , within the range specified.  $C_F$  is recommended for noise management, with values that depend on the application.

# ACS712

## Fully Integrated, Hall Effect-Based Linear Current Sensor with 2.1 kVRMS Voltage Isolation and a Low-Resistance Current Conductor

### Description (continued)

loss. The thickness of the copper conductor allows survival of the device at up to 5× overcurrent conditions. The terminals of the conductive path are electrically isolated from the sensor leads (pins 5 through 8). This allows the ACS712 current sensor to be used in applications requiring electrical isolation without the use of opto-isolators or other costly isolation techniques.

The ACS712 is provided in a small, surface mount SOIC8 package. The leadframe is plated with 100% matte tin, which is compatible with standard lead (Pb) free printed circuit board assembly processes. Internally, the device is Pb-free, except for flip-chip high-temperature Pb-based solder balls, currently exempt from RoHS. The device is fully calibrated prior to shipment from the factory.

### Selection Guide

Part Number	Packaging*	T <sub>A</sub> (°C)	Optimized Range, I <sub>P</sub> (A)	Sensitivity, Sens <sub>(typ)</sub> (mV/A)
ACS712ELCTR-05B-T	Tape and reel, 3000 pieces/reel	-40 to 85	±5	185
ACS712ELCTR-20A-T	Tape and reel, 3000 pieces/reel	-40 to 85	±20	100
ACS712ELCTR-30A-T	Tape and reel, 3000 pieces/reel	-40 to 85	±30	66

\*Contact Allegro for additional packing options.

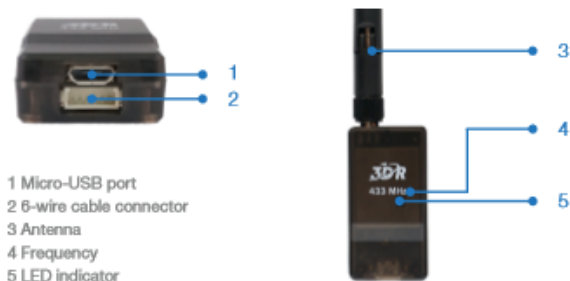
### Absolute Maximum Ratings

Characteristic	Symbol	Notes	Rating	Units
Supply Voltage	V <sub>CC</sub>		8	V
Reverse Supply Voltage	V <sub>RECC</sub>		-0.1	V
Output Voltage	V <sub>OUT</sub>		8	V
Reverse Output Voltage	V <sub>REOUT</sub>		-0.1	V
Reinforced Isolation Voltage	V <sub>ISO</sub>	Pins 1-4 and 5-8; 60 Hz, 1 minute, T <sub>A</sub> =25°C Voltage applied to leadframe (tp+ pins), based on IEC 60950	2100	V
Basic Isolation Voltage	V <sub>ISO(BAS)</sub>	Pins 1-4 and 5-8; 60 Hz, 1 minute, T <sub>A</sub> =25°C Voltage applied to leadframe (tp+ pins), based on IEC 60950	1500	V
Output Current Source	I <sub>OUT(SOURCE)</sub>		3	mA
Output Current Sink	I <sub>OUT(SINK)</sub>		10	mA
Overcurrent Transient Tolerance	I <sub>P</sub>	1 pulse, 100 ms	100	A
Nominal Operating Ambient Temperature	T <sub>A</sub>	Range E	-40 to 85	°C
Maximum Junction Temperature	T <sub>J(max)</sub>		165	°C
Storage Temperature	T <sub>STG</sub>		-65 to 170	°C

Parameter	Specification
Fire and Electric Shock	CAN/CSA-C22.2 No. 60950-1-03 UL 60950-1:2003 EN 60950-1:2001

## Lampiran C Datasheet 3DR Telemetry Module 915MHZ

### RADIO DESCRIPTION



### SPECIFICATIONS

#### Processing

100 mW maximum output power (adjustable)  
-117 dBm receive sensitivity  
Based on HopeRF's HM-TRP module  
RP-SMA connector  
2-way full-duplex communication through adaptive TDM  
UART interface  
Transparent serial link  
MAVLink protocol framing  
Frequency Hopping Spread Spectrum (FHSS)  
Configurable duty cycle  
Error correction corrects up to 25% of bit errors  
Open-source SIK firmware  
Configurable through Mission Planner & APM Planner

#### Features

Interchangeable air and ground modules  
915 or 433 mHz  
Micro-USB port  
6-position DF13 connector

#### Dimensions

26.7 cm x 55.5 cm x 13.3 cm  
(without antenna)

#### Power

Supply voltage: 3.7-6 VDC (from USB or DF13)  
Transmit current: 100 mA at 30 dBm  
Receive current: 25 mA  
Serial interface: 3.3 V UART

### SUPPORT

For more information about mission planner applications and APM firmware, visit [ardupilot.com](http://ardupilot.com). For online documentation of 3DR Radios, visit [goo.gl/Tsrksf](http://goo.gl/Tsrksf).

For customer support, contact us at [help@3drobotics.com](mailto:help@3drobotics.com) or call our support line at +1 (858) 225-1414.

## Lampiran D Library Mbed.h

```
/* mbed Microcontroller Library
 * Copyright (c) 2006-2013 ARM Limited
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software
 * distributed under the License is distributed on an "AS IS"
 * BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied.
 * See the License for the specific language governing permissions
 * and
 * limitations under the License.
 */
#ifndef MBED_H
#define MBED_H

#define MBED_LIBRARY_VERSION 137

#if MBED_CONF_RTOS_PRESENT
// RTOS present, this is valid only for mbed OS 5
#define MBED_MAJOR_VERSION 5
#define MBED_MINOR_VERSION 3
#define MBED_PATCH_VERSION 6
#else
// mbed 2
#define MBED_MAJOR_VERSION 2
#define MBED_MINOR_VERSION 0
```

```

#define MBED_PATCH_VERSION
MBED_LIBRARY_VERSION
#endif

#define MBED_ENCODE_VERSION(major, minor, patch)
((major)*10000 + (minor)*100 + (patch))

#define MBED_VERSION
MBED_ENCODE_VERSION(MBED_MAJOR_VERSION,
MBED_MINOR_VERSION, MBED_PATCH_VERSION)

#if MBED_CONF_RTOS_PRESENT
#include "rtos/rtos.h"
#endif

#if MBED_CONF_NSAPI_PRESENT
#include "netsocket/nsapi.h"
#endif

#if MBED_CONF_EVENTS_PRESENT
#include "events/mbed_events.h"
#endif

#include "platform/toolchain.h"
#include "platform/platform.h"

// Useful C libraries
#include <math.h>
#include <time.h>

// mbed Debug libraries
#include "platform/mbed_error.h"
#include "platform/mbed_interface.h"
#include "platform/mbed_assert.h"

// mbed Peripheral components
#include "drivers/DigitalIn.h"

```

```
#include "drivers/DigitalOut.h"
#include "drivers/DigitalInOut.h"
#include "drivers/BusIn.h"
#include "drivers/BusOut.h"
#include "drivers/BusInOut.h"
#include "drivers/PortIn.h"
#include "drivers/PortInOut.h"
#include "drivers/PortOut.h"
#include "drivers/AnalogIn.h"
#include "drivers/AnalogOut.h"
#include "drivers/PwmOut.h"
#include "drivers/Serial.h"
#include "drivers/SPI.h"
#include "drivers/SPISlave.h"
#include "drivers/I2C.h"
#include "drivers/I2CSlave.h"
#include "drivers/Ethernet.h"
#include "drivers/CAN.h"
#include "drivers/RawSerial.h"
```

```
// mbed Internal components
#include "drivers/Timer.h"
#include "drivers/Ticker.h"
#include "drivers/Timeout.h"
#include "drivers/LowPowerTimeout.h"
#include "drivers/LowPowerTicker.h"
#include "drivers/LowPowerTimer.h"
#include "drivers/LocalFileSystem.h"
#include "drivers/InterruptIn.h"
#include "platform/wait_api.h"
#include "hal/sleep_api.h"
#include "platform/rtc_time.h"
```

```
// mbed Non-hardware components
#include "platform/Callback.h"
#include "platform/FunctionPointer.h"
```

```
using namespace mbed;  
using namespace std;
```

## **Lampiran E RTOS.h**

```
/** \addtogroup rtos */  
/** @ { */  
/* mbed Microcontroller Library  
 * Copyright (c) 2006-2012 ARM Limited  
 *  
 * Permission is hereby granted, free of charge, to any person  
obtaining a copy  
 * of this software and associated documentation files (the  
"Software"), to deal  
 * in the Software without restriction, including without limitation  
the rights  
 * to use, copy, modify, merge, publish, distribute, sublicense,  
and/or sell  
 * copies of the Software, and to permit persons to whom the  
Software is  
 * furnished to do so, subject to the following conditions:  
 *  
 * The above copyright notice and this permission notice shall be  
included in  
 * all copies or substantial portions of the Software.  
 *  
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT  
WARRANTY OF ANY KIND, EXPRESS OR  
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE  
WARRANTIES OF MERCHANTABILITY,  
 * FITNESS FOR A PARTICULAR PURPOSE AND  
NONINFRINGEMENT. IN NO EVENT SHALL THE  
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR  
ANY CLAIM, DAMAGES OR OTHER  
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT,  
TORT OR OTHERWISE, ARISING FROM,
```



```
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR  
THE USE OR OTHER DEALINGS IN THE  
* SOFTWARE.
```

```
*/  
#ifndef RTOS_H  
#define RTOS_H
```

```
#include "rtos/Thread.h"  
#include "rtos/Mutex.h"  
#include "rtos/RtosTimer.h"  
#include "rtos/Semaphore.h"  
#include "rtos/Mail.h"  
#include "rtos/MemoryPool.h"  
#include "rtos/Queue.h"
```

```
using namespace rtos;
```

```
/* Get mbed lib version number, as RTOS depends on mbed lib  
features
```

```
    like mbed_error, Callback and others.
```

```
*/  
#include "mbed.h"
```

```
#if (MBED_LIBRARY_VERSION < 122)  
#error "This version of RTOS requires mbed library version > 121"  
#endif
```

```
#endif
```

```
/** @ */
```

## **Lampiran F Library SDFile.h**

/\* mbed Microcontroller Library

\* Copyright (c) 2006-2012 ARM Limited

\*

\* Permission is hereby granted, free of charge, to any person obtaining a copy

\* of this software and associated documentation files (the "Software"), to deal

\* in the Software without restriction, including without limitation the rights

\* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

\* copies of the Software, and to permit persons to whom the Software is

\* furnished to do so, subject to the following conditions:

\*

\* The above copyright notice and this permission notice shall be included in

\* all copies or substantial portions of the Software.

\*

\* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

\* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

\* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

\* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

\* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

\* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

\* SOFTWARE.

\*/

#ifndef MBED\_SDFILESYSTEM\_H

#define MBED\_SDFILESYSTEM\_H

```

#include "mbed.h"
#include "FATFileSystem.h"
#include <stdint.h>

/** Access the filesystem on an SD Card using SPI
 *
 * @code
 * #include "mbed.h"
 * #include "SDFileSystem.h"
 *
 * SDFileSystem sd(p5, p6, p7, p12, "sd"); // mosi, miso, sclk, cs
 *
 * int main() {
 *     FILE *fp = fopen("/sd/myfile.txt", "w");
 *     fprintf(fp, "Hello World!\n");
 *     fclose(fp);
 * }
 */
class SDFileSystem : public FATFileSystem {
public:

    /** Create the File System for accessing an SD Card using SPI
     *
     * @param mosi SPI mosi pin connected to SD Card
     * @param miso SPI miso pin connected to SD Card
     * @param sclk SPI sclk pin connected to SD Card
     * @param cs DigitalOut pin used as SD Card chip select
     * @param name The name used to access the virtual filesystem
     */
    SDFileSystem(PinName mosi, PinName miso, PinName sclk,
PinName cs, const char* name);
    virtual int disk_initialize();
    virtual int disk_status();
    virtual int disk_read(uint8_t* buffer, uint32_t block_number,
uint32_t count);
    virtual int disk_write(const uint8_t* buffer, uint32_t
block_number, uint32_t count);

```

```
virtual int disk_sync();
virtual uint32_t disk_sectors();
```

protected:

```
int _cmd(int cmd, int arg);
int _cmdx(int cmd, int arg);
int _cmd8();
int _cmd58();
int initialise_card();
int initialise_card_v1();
int initialise_card_v2();
```

```
int _read(uint8_t * buffer, uint32_t length);
int _write(const uint8_t *buffer, uint32_t length);
uint32_t _sd_sectors();
uint32_t _sectors;
```

```
void set_init_sck(uint32_t sck) { _init_sck = sck; }
```

// Note: The highest SPI clock rate is 20 MHz for MMC and 25 MHz for SD

```
void set_transfer_sck(uint32_t sck) { _transfer_sck = sck; }
uint32_t _init_sck;
uint32_t _transfer_sck;
```

```
SPI _spi;
DigitalOut _cs;
int cdv;
int _is_initialized;
};
```

```
#endif
```

## BIODATA PENULIS



Penulis yang bernama Ergi Qibar Nugraha dilahirkan di Magetan pada tanggal 17 Agustus 1997. Penulis merupakan anak pertama dari dua bersaudara dari orang tua Bapak Nugroho dan Ibu uniek. Penulis menempuh pendidikan di SMA 8 Surabaya Kemudian penulis melanjutkan ke jenjang yang lebih tinggi yait di Institut Teknologi Sepuluh Nopember Surabaya, tepatnya di Departemen Teknik Instrumentasi Fakultas Vokasi di tahun 2015 dan terdaftar dengan NRP 10 51 15 000 00

078.

Di Departemen Teknik Instrumentasi ini penulis aktif di beberapa organisasi kemahasiswaan dan minat bakat yaitu menjadi staf UKM Bola Basket ITS, dan menjadi pemain basket di TIM BOLA BASKET ITS, penulis telah melaksanakan Tugas Akhir Program Studi DIII Teknologi Instrumentasi Departemen Teknik Instrumentasi FV ITS.

Selesainya Tugas Akhir ini diajukan penulis untuk memenuhi syarat memperoleh gelar Ahli Madya di Departemen Teknik Instrumentasi Fakultas Vokasi Institut Teknologi Sepuluh Nopember Surabaya.

Ergi.@gmail.com